

Opinnäytetyö (AMK)

Tietotekniikan koulutusohjelma

Mediatekniikka

2014

Antti Talvitie

# RESPONSIIVINEN VERKKSUUNNITTELU JA SEN HYÖDYNTÄMINEN SISÄLLÖNHALLINTA- JÄRJESTELMÄSSÄ



TURUN AMMATTIKORKEAKOULU  
TURKU UNIVERSITY OF APPLIED SCIENCES

Antti Talvitie

# RESPONSIIVINEN SUUNNITTELU JA SEN HYÖDYNTÄMINEN SISÄLLÖNHALLINTAJÄRJESTELMÄSSÄ

Verkkosivut elävät jatkuvassa murroksessa. Uusia päätelaitteita saapuu markkinoille nopealla syklillä ja sivujen pitäisi mukautua näiden laitteiden näytöille. Parhaiten siinä onnistutaan tällä hetkellä tarjoamalla käyttäjälle responsiivisia verkkosivuja. Responsiivisilla verkkosivuilla tarkoitetaan sivuja, jotka skaalautuvat päätelaitteen mukaan.

Responsiivisen suunnittelun tueksi kehitetään koko ajan uusia tekniikoita ongelmakohtien ratkaisuksi ja sisällön optimoimiseksi. Alaa on seurattava jatkuvasti, ja on tärkeää omaksua uudet ideat ja ajatusmallit nopeasti.

Opinnäytetyössä tarkasteltiin responsiivista verkkosuunnittelua ja sen käyttöönottoa sisällönhallintajärjestelmässä. Työssä tutkittiin responsiivisen verkkosuunnittelun peruseriä ja miten niitä sovelletaan käytännössä. Työtä varten suunniteltiin responsiivinen pohja, jossa tekniikoita hyödynnettiin. Pohjaa testattiin Concrete5-sisällönhallintajärjestelmässä. Työn tavoitteena oli perehtyä syvemmin responsiivisen verkkosuunnittelun tekniikoihin ja kokeilla niitä myös käytännössä.

Opinnäytetyö onnistui hyvin ja asetetut tavoitteet saavutettiin. Työtä varten suunniteltu responsiivinen pohja ei päädy tuotantokäyttöön, mutta sitä voi jatkossa käyttää mahdollisena testialustana. Työ vahvisti omaa käsitystä siitä, että responsiivinen verkkosuunnittelu on tällä hetkellä järkevin tapa toteuttaa mahdollisimman monipuolinen ja suurinta osaa käyttäjistä palveleva verkkosivu.

## ASIASANAT:

HTML, CSS, sisällönhallintajärjestelmä, responsiivinen verkkosuunnittelu

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Information Technology | Media Technology

December 2014 | 28

Instructor: Principal Lecturer, Ph. D. Mika Luimula

Antti Talvitie

# RESPONSIVE WEB DESIGN AND ITS USAGE IN A CONTENT MANAGEMENT SYSTEM

Websites are living in constant transition. New devices are brought to the market at a rapid rate and the websites should be compatible with these new devices. At the moment the best solution is to offer responsive websites to the users. Responsive web design means that the site is designed to scale to the resolution of the users' device.

New techniques are developed all the time to support and to solve the problems in responsive web design. The development of this field of expertise must be followed constantly and one must adopt new techniques and ideas quickly to keep up with the date.

This thesis examines responsive web design and how it can be used in a content management system. This work explored the basic idea behind responsive web design and how it can be used in practical implementation. The practical part of the thesis demonstrates a simple responsive base theme and how it has been enabled in a content management system called Concrete5. The main objectives of this thesis were to further explore responsive web design and try it in practice.

The result of thesis was successful and all the objectives were achieved. In the future the responsive base theme can be used as a test ground for new kinds of technical approaches. This thesis strengthened my own opinion that responsive web design is the most practical way to create websites that serve the vast majority of users.

## KEYWORDS:

HTML, CSS, content management system, responsive web design

# SISÄLTÖ

<b>SANASTO</b>	<b>6</b>
<b>1 JOHDANTO</b>	<b>7</b>
<b>2 RESPONSIIVINEN VERKKOSUUNNITTELU</b>	<b>8</b>
2.1 Responsiivisen verkkosuunnittelun hyödyt	9
2.2 Responsiivisen verkkosuunnittelun haitat	9
2.3 Sivurakenne	10
2.4 Navigaatio	10
2.5 Typografia	12
2.6 Kuvat	14
<b>3 SISÄLLÖNHALLINTAJÄRJESTELMÄ</b>	<b>16</b>
3.1 Wordpress	16
3.2 Drupal	17
3.3 Joomla	17
3.4 Concrete5	18
3.5 Julkaisujärjestelmän valinta	18
<b>4 CONCRET5:N ASENNUS</b>	<b>19</b>
<b>5 RESPONSIIVISEN POHJAN HYÖDYNTÄMINEN</b>	
<b>SISÄLLÖNHALLINTAJÄRJESTELMÄSSÄ</b>	<b>22</b>
5.1 Toimeksiannon tavoite	22
5.2 Responsiivinen pohja	22
5.2.1 Ylätunniste	25
5.2.2 Sisältöalue	25
5.2.3 Alatunniste	25
<b>6 YHTEENVETO</b>	<b>27</b>
<b>LÄHTEET</b>	<b>28</b>

# KUVAT

Kuva 1. Esimerkki Rock Hammer –verkkosivun navigaatiosta	11
Kuva 2. Esimerkki Rock Hammer –verkkosivun suljetusta navigaatiosta alle 675 pikseliä leveillä näytöillä	11
Kuva 3. Esimerkki Rock Hammer –verkkosivun avatusta navigaatiosta alle 675 pikseliä leveillä näytöillä	12
Kuva 4. Esimerkki typografian hyödyntämisestä responsiivisella sivulla	13
Kuva 5. Tyhjä tietokanta sisällönhallintajärjestelmää varten	19
Kuva 6. Kielivalinta	19
Kuva 7. Vaatimusten testaus	20
Kuva 8. Sivun, tietokannan ja ylläpitäjän tiedot	21
Kuva 9. Responsiivinen pohja yli 768 pikseliä leveillä näytöillä	23
Kuva 10. Responsiivinen pohja alle 768 pikseliä leveillä näytöillä.	23
Kuva 11. Teemapohjan aktivointi	26

# SANASTO

Apache	Avoimeen lähdekoodiin perustuva palvelinohjelma.
CSS	WWW-dokumenttien tyylittelyissä käytetty merkkaustapa.
em	Verkkosuunnittelussa käytetty suhteellinen kokomäärittely.
HTML-dokumentti	tekstitiedosto, jossa määritellään esimerkiksi verkkosivun rakenne.
jQuery	Etenkin verkkosivuilla käytetty JavaScript-kirjasto.
MariaDB	Relaatiotietokantajärjestelmä.
Microsoft IIS	
MySQL	Avoimeen lähdekoodiin perustuva relaatiotietokantajärjestelmä.
Nginx	Avoimeen lähdekoodiin perustuva palvelinohjelma.
PHP	Etenkin verkkosivuilla käytetty ohjelmointikieli.
PostgreSQL	Avoimeen lähdekoodiin perustuva tietokantajärjestelmä.
rem	Verkkosuunnittelussa käytetty suhteellinen kokomäärittely.
SQLite	Relaatiotietokantajärjestelmä.

# 1 JOHDANTO

Internetin räjähdysmäinen kasvu viimeisen vuosikymmenen aikana on muuttanut koko tavan selata verkkosivuja.

Samalla kun mobiililaitteiden myynti kasvaa, yleistyy myös niiden käyttö. Mobiililaitteiden korvautessa tavanomaisen tietokoneen, muuttuu myös verkkosivujen standardit. Nykypäivän verkkosivun tulisi toimia moitteetta päätelaitteen ruudun koosta huolimatta.

Käytännössä vaihtoehtoja on tällä hetkellä kaksi – erikseen mobiililaitteelle tehty sivusto ja responsiivinen sivusto. Molempia tekniikoita käytetään, ja molemmissa on omat hyvät ja huonot puolensa. Responsiivisen sivuston hyviin puoliin lukeutuu sen toimiminen useille eri päätelaitteilla. Hakukoneoptimointi on myös helpompi ja tehokkaampaa tehdä ainoastaan yhdelle verkkosivulle, kuin kahdelle eri versiolle.

Tämä opinnäytetyö käsittelee responsiivista suunnittelua ja sen hyödyntämistä sisällönhallintajärjestelmässä. Opinnäytetyössä tutustutaan responsiivisen suunnittelun ja Concrete5:n peruseräkkeisiin. Käytännön osuudessa käydään läpi Concrete5:n asennus ja responsiivisen pohjan käyttöönotto, mikä on toimeksiantajalta saatu tehtävä.

## 2 RESPONSIIVINEN VERKKOSUUNNITTELU

Ennen älypuhelimia ja tabletteja internetiä selattiin kannettavilla- ja pöytäkoneilla. Verkkosivujen mitat olivat yleisesti kiinteät, ja esimerkiksi 960 pikseliä leveä sivu näkyi oikein lähes kaikilla laitteilla. Älypuhelimien ja tablettien tultua markkinoille verkkosivujen suunnittelu koki suuren muutoksen. Käyttäjän laitteen mukaan mukautuvasta pohjasta kirjoitti ensimmäisen kerran Cameron Adams vuonna 2004. [1] Vuonna 2010 Ethan Marcotte käytti ensimmäisen kerran termiä Responsive web design artikkelissa A List Apart [2].

Vuoden 2004 jälkeen internet ja verkkosivut ovat kokeneet suuren muutoksen ja ala on jatkuvassa murroksessa. Kansainvälisen ICT-alan tutkimus- ja konsultointiyritys Gartnerin mukaan perinteisten tietokoneiden myynti jatkaa laskuaan ja mobiililaitteiden myynti on selvässä nousussa. [3]

Responsiivisen verkkosuunnittelun ideana on skaalata verkkosivun sisältö käyttäjän laitteen ruutukoon mukaan. Responsiivisella sivulla pystytään tarjoamaan käyttäjälle mahdollisimman yhtenäinen kokonaisuus päätelaitteen ruutukoosta huolimatta. Responsiivisuuden idea perustuu prosentuaalisiin mittoihin. Ennen sivu määriteltiin esimerkiksi 960 pikseliä leveäksi, nykyisin sivun leveydeksi voidaan asettaa 100 %. Ruutukoon muuttuessa sivun elementit skaalautuvat prosentuaalisesti.

Pikseleiden muuttaminen prosenteiksi ei tee sivusta responsiivista. Mobiiliselaimia varten verkkosivun lähdekoodiin pitää sisällyttää viewport-elementti. Viewportin avulla pystytään esimerkiksi tunnistamaan laitteen resoluutio ja asettamaan se verkkosivun maksimileveydeksi. [4]

Esimerkkikoodissa name-kohta aktivoi viewportin, content-kohtaan on määritetty viewportin leveydeksi laitteen ruudun leveys ja user-scalable-kohdassa zoomaus on poistettu käytöstä.

```
<meta name="viewport" content="width=device-width, user-scalable=no">
```



## 2.1 Responsiivisen verkkosuunnittelun hyödyt

Responsiivisen suunnittelun suurin hyöty on ainoastaan yhden sivuston suunnittelu. Päivitysten hallinta on myös yksinkertaisempaa yhden version kanssa, kuin esimerkiksi pöytäkoneille tarkoitetun ja erillisen mobiilisivun kombinaatiolla.

Etenkin yritysten sivujen on tärkeää löytyä hakukoneilla ja esimerkiksi Google, joka on yksi maailman suurimmista ja suosituimmista hakukoneista, suosii responsiivisia sivuja. Yrityksen perusteluna on se, että sama verkkosivu pitäisi olla saatavana kaikille laitteesta riippumatta. [5]

Responsiivisen verkkosivun hakukoneoptimointi on myös huomattavasti tehokkaampaa ja helpompaa verrattaessa esimerkiksi erillisen mobiilisivun optimointiin. Hyvin suunnitellulla sivulla url-rakenne pysyy samana ja näin hakukoneet pystyvät paremmin indeksoimaan sivun sisältöä.

## 2.2 Responsiivisen verkkosuunnittelun haitat

Responsiivisesti suunnitelluilla verkkosivuilla on luonnollisesti myös haittapuolet ja omat haasteensa. Hyvin suunnitellun responsiivisen verkkosivun toteuttaminen vie enemmän aikaa, kuin pelkän desktop-sivun rakentaminen.

Vanhemmat selaimet, kuten Internet Explorer 8 ja sitä vanhemmat, eivät kunnolla tue kaikkia responsiivisessa suunnittelussa käytettyjä tekniikoita, jolloin verkkosivu saattaa näkyä näillä selaimilla huonosti tai ei ollenkaan. Näitä selaimia ei juurikaan enää käytetä yleisesti, mutta on kuitenkin hyvä tiedostaa, että vanhempiakin selaimia on edelleen käytössä. [6]

Responsiiviset sivut tarjoavat saman sisällön ruudun koosta tai päätelaitteesta riippumatta. Tämä tarkoittaa, että esimerkiksi älypuhelimella joutuu lataamaan saman määrän tietoa, kuin pöytäkoneella.

Responsiivisessa suunnittelussa on omat sudenkuoppansa, mutta hyvällä suunnittelulla ja tehokkaalla sisällön optimoinnilla uskoisin, että se on kuitenkin järkevin tekniikka toteuttaa verkkosivuja tällä hetkellä.

## 2.3 Sivurakenne

HTML-dokumentin rakenne on yksinkertaisuudessaan seuraava:

```
<html>
<head>
<title></title>
</head>
<body>
</body>
</html>
```

HTML-dokumentti alkaa ja loppuu html-tagilla. HTML-tagia seuraa head-osio, jossa voidaan kutsua mm. tyylitiedostoja tai esimerkiksi sivuilla käytettävää JavaScript koodia. Pakollinen title-tagin määrittää sivun otsikon. Head-alueen sisältö ei näy lainkaan verkkosivulla. Body-tagien sisällä sijaitsee sivun varsinainen sisältö ja sivurakenne. [7]

## 2.4 Navigaatio

Verkkosivun navigaation tulisi olla heti näkyvissä ja sitä ei pitäisi joutua etsimään sivulta. [8]

Navigaation löytää verkkosivuilta yleensä header-elementin sisältä, ja se sijaitsee hyvin usein sivun yläosassa. Käytettävyyden kannalta sen sijainti tulisi säilyttää samana kautta sivuston. Päänavigaation linkkien nimeämisessä kannattaa myös käyttää yhtenäistä logiikkaa ja linjaa. Valikkorakenteen suunnittelu hankaloituu isoissa sivustoissa, joissa sivumäärä on iso ja pääsivut

sisältävät alasivuja. Valikkorakenne muotoillaan melko usein vierekkäisiksi linkeiksi (Kuva 1). Navigaatio toimii samalla sivun sisällysluettelona ja runkona.



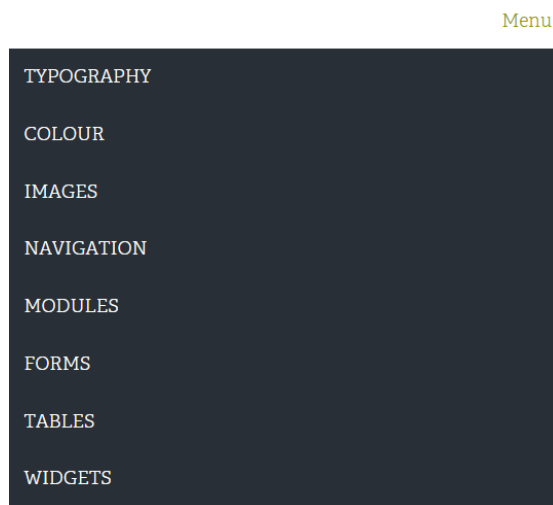
Kuva 1. Esimerkki Rock Hammer –verkkosivun navigaatiosta

Pienillä näytöillä tavanomainen navigaatio on haastavaa saada mahtumaan ruudulle. Esimerkiksi älypuhelimien ruudulle ei yksinkertaisesti mahdu kovin laajaa, selkeää ja yhdenmukaista valikkorakennetta. Alle 675 pikseliä leveillä näytöillä valikkorakenne on pienennetty, ja se saadaan avattua menu-painiketta painamalla (Kuva 2).



Kuva 2. Esimerkki Rock Hammer –verkkosivun suljetusta navigaatiosta alle 675 pikseliä leveillä näytöillä

Esimerkkisivun valikko löytyy oikeasta yläkulmasta, Menu-painiketta painamalla (Kuva 3).



Kuva 3. Esimerkki Rock Hammer -verkkosivun avatusta navigaatiosta alle 675 pikseliä leveillä näytöillä

## 2.5 Typografia

Ennen responsiivisen suunnittelun yleistymistä myös typografian mittayksikkönä käytettiin pikseleitä. Tavallisesti verkkosuunnittelussa annetaan jokaiselle tekstityylille omat arvonsa mm. tekstin koolle ja marginaalille. Ruudun koon muuttuessa nämä arvot pitää muuttaa kulloinkin sopivaksi, jolloin pikselikoon määrittäminen erikseen jokaiselle tekstityypille on todella työlästä. Responsiivisen suunnittelun myötä suunnittelijat ovat alkaneet käyttää em- ja rem-yksikköä.

Em-yksikkö on joustava ja suhteellinen koko. Mikäli verkkosivun body-elementissä on määritelty, että fontin koko on 14 pikseliä, 1 em vastaa 14:ää pikseliä. Em-yksikkö voidaan määritellä erikseen eri elementeille. Rem-yksikkö tulee sanasta root em, ja se voidaan määritellä html-elementissä, joka kattaa koko sivuston. [9] Toisin sanoen em-yksiköllä määritelty tekstin koko riippuu elementistä, jonka sisällä tekstisisältö on. Rem-yksikkö sen sijaan pysyy saman kokoisena koko sivun alueella.

Responsiivisessa suunnittelussa rem-yksikköä voidaan hyödyntää esimerkiksi kuvassa 4 esitetyllä tavalla.

```
1  html {  
2    font-size: 14px;  
3  }  
4  @media screen and (max-width: 480px) {  
5    html {  
6      font-size: 16px;  
7    }  
8  }  
9  @media screen and (max-width: 768px) {  
10   html {  
11     font-size: 18px;  
12   }  
13 }  
14 @media screen and (max-width: 1024px) {  
15   html {  
16     font-size: 22px;  
17   }  
18 }  
19 h1 {  
20   font-size: 2rem;  
21 }  
22 h2 {  
23   font-size: 1.5rem;  
24 }  
25 p {  
26   font-size: 1rem;  
27 }
```

Kuva 4. Esimerkki typografian hyödyntämisestä responsiivisella sivulla

Esimerkin tekstikoot on määritelty seuraavasti: H1-elementti 2 rem, H2-elementti 1,5 rem ja p-elementti 1 rem. HTML-dokumentin vakiotekstikoko on 14 pikseliä. Näin ollen H1-elementti on kooltaan 28 pikseliä, H2-elementti 21 pikseliä ja p-elementti on 14 pikseliä. Esimerkissä on käytetty kolmea eri breakpoint-leveyttä: alle 1 024 pikseliä, alle 768 pikseliä ja alle 480 pikseliä. Näiden leveyksien tullessa voimaan HTML-dokumentin vakiotekstikoko vaihtuu alle 1 024 leveillä näytöillä 22 pikseliin, alle 768 pikseliä leveillä 18 pikseliin ja alle 480 pikseliä leveillä 16 pikseliin. Rem-arvoa ja mediaqueryä hyödyntämällä teksti saadaan skaalattua eri ruutukokojen mukaan muuttamalla ainoastaan dokumentin vakiotekstikokoa.

## 2.6 Kuvat

Kuvien käyttö verkkosivuilla, kuten melkein missä tahansa esitysmuodossa, on tehokas apuväline. Etenkin viime vuosina verkkosivuilla on käytetty yhä isompia ja tarkempia kuvia mahdollisimman hienon visuaalisen ilmeen toteuttamiseksi. Luonnollisesti kuvien on tarkoitus palvella tarkoitusta ja turhien kuvien lisääminen sivuille voi pahimmillaan rikkoa kokonaiskuvan ja käyttökokemuksen. Esimerkiksi Nielsen Norman Groupin tekemän tutkimuksen mukaan käyttäjä saattoi käyttää 10 % enemmän aikaa kuvien katsomiseen verrattuna vieressä olevaan tekstiin. [10]

Kuvat vievät enemmän tilaa palvelimelta, kuin esimerkiksi tekstisisältö. Responsiivisessa suunnittelussa tämä seikka korostuu entisestään, koska pienemmät päätelaitteet eivät vaadi yhtä isoja ja tarkkoja kuvia, kuin esimerkiksi suuri tietokoneen näyttö. Responsiivisessa suunnittelussa ei ole yhtä oikeaa standardia kuvien käsittelyyn, vaan käytettävissä olevista ratkaisuista on valittava paras mahdollinen tilanteen mukaan.

Yksinkertaisin tapa skaalata kuva oikean kokoiseksi on määrittää kuva-elementin leveys sataprosenttiseksi. Näin kuva täyttää automaattisesti koko responsiivisen elementin, johon ne on asetettu. Näyttökoon muuttuessa kuvan koko muuttuu responsiivisen elementin mukana. Datakäytön kannalta tämä on kuitenkin huono tapa ja sivun lataaminen vie enemmän aikaa hitaammilla yhteyksillä. Kuvan koko muuttuu ainoastaan ruudulla, mutta kuvalähde pysyy samana, jolloin pienemällä näyttöpäätteellä selaava lataa saman kuvan, kuin isolla näytöllä selaava.

Kuvien lataamiseen ja esittämiseen verkkosivulla löytyy useampia eri lisäosia ja ratkaisuja. HiSRC on yksinkertainen jQuerya hyödyntävä lisäosa, jonka avulla verkkosivuille voidaan ladata useampi versio kuvasta ja HiSRC:n avulla kuvasta ladataan aina sopivin versio.

Käytännössä HiSRC testaa verkon nopeuden ja päättää sen avulla onko käytössä mahdollisesti hidas mobiililaitte vai nopeampi yhteys. Ensin kuvasta

latautuu kaikkein heikkolaatuisin versio, ja mikäli yhteys on riittävän nopea kuvasta ladataan parempilaatuinen versio. [11]

Huonona puolena HiSRC:ssä on se, että kuvasta latautuu nopealla yhteydellä kaksi eri versiota. Hitaammalla yhteydellä kuvasta saadaan toisaalta heti käsittelyyn optimoitu versio. Tässäkin menetelmässä on omat huonot ja hyvät puolensa. Eri vaihtoehtoja löytyy monia ja jokaisen projektin kohdalla kannattaakin tutustua eri menetelmiin ja valita tapauskohtaisesti paras mahdollinen ratkaisu responsiivisten kuvien toteuttamiseksi.

### 3.5 Media queryt

CSS3:n mukana tuli uusi työkalu verkkosivujen kehittämiseen – media queryt.[12] Responsiivisessa suunnittelussa media queryitä käytetään pääsääntöisesti laitteen resoluution selvittämiseen, mikä mahdollistaa sivuston suunnittelun erikokoisille näytöille. Kohtia, joissa media queryn asettamat säännöt astuvat voimaan, kutsutaan breakpointeiksi. Breakpointit määritellään pikseleinä ja näiden ns. murtumakohtien suunnitteluun vaikuttaa ensisijaisesti päätelaitteiden resoluutiot. Myös verkkosivun ulkoasu ja asettelu pitää huomioida murtumakohtia valittaessa. Älypuhelimien ja tablettien tekniikka kehittyy huimaa vauhtia ja samalla näyttöjen tarkkuudet paranevat.

Esimerkki media queryn käytöstä:

```
@media screen and (max-width: 768px) {
  .example {
    width: 100%;
  }
}
```

Esimerkissä alle 768 pikseliä leveillä ruuduilla example-luokan elementtien leveydeksi asetetaan 100 %.

### 3 SISÄLLÖNHALLINTAJÄRJESTELMÄ

Sisällönhallintajärjestelmällä tarkoitetaan järjetelemää, jonka avulla sisältöä pystyy julkaisemaan, muokkaamaan, järjetelemään ja ylläpitämään [13].

Sisällönhallintajärjestelmän hyödyllisyys tulee esiin etenkin sellaisissa kokonaisuuksissa, joissa tietoa on paljon, käyttäjiä useita ja päivitystiheys nopeaa. Sisällönhallintajärjestelmällä tietoa pystytään käsittelemään keskitetysti vaikka käyttäjiä olisikin useampi. Dataa on mahdollista esittää useassa eri muodossa ja sitä pystytään jakamaan muiden rajapintojen kanssa.

Tällä hetkellä W3Techs-internetsivun mukaan kolme yleisintä avoimen lähdekoodin sisällönhallintajärjestelmää on Wordpress, Joomla ja Drupal. Wordpressiä käyttää hetkellä 23 %, Joomlaan 3 % ja Drupal on käytössä 2 %:lla. Concrete5:n osuus on verrattaen pieni. Sen käyttöaste on ainoastaan 0,1 %. [14]

#### 3.1 Wordpress

Wordpress on avoimeen lähdekoodiin perustuva sisällönhallintajärjestelmä, joka julkaistiin vuonna 2003. Alun perin Wordpress luotiin blogialustaksi, mutta se on sittemmin kasvanut täysiveriseksi ja suosituimmaksi sisällönhallintajärjestelmäksi internetissä. Wordpressin käyttö vaatii PHP-ohjelmointikieltä, MySQL-tietokannan ja palvelimen pitää olla Apache- tai Nginx-pohjainen. [15]

Wordpressin etuja on suuren käyttäjäryhmän ansiosta suuri tukiverkko ja laaja valikoima ilmaisia lisäosia. Wordpressin asennus ja käyttö on tehty myös todella helpoksi ja sen oppimiskäyrä on paljon loivempi, kuin esimerkiksi Drupalin ja Joomlaan. Suuri määrä ilmaisia lisäosia asettaa Wordpressin toisaalta myös hyvin haavoittuvaksi ja epävarmaksi sisällönhallintajärjestelmäksi. Wordpress soveltuu parhaiten hieman yksinkertaisempien ja pienempien sivustojen sisällönhallintajärjestelmäksi.



### 3.2 Drupal

Wordpressin tapaan myös Drupal perustuu avoimeen lähdekoodiin. Drupal julkaistiin virallisesti vuonna 2001, mutta alustan kehitystyö aloitettiin jo vuonna 1999. [16]

Drupal on erittäin monipuolinen ja joustava sisällönhallintajärjestelmä. Drupal vaatii palvelimelta Apachen, Nginxin tai Microsoft IIS:n. Tietokannan osalta Drupal on Wordpressiä joustavampi. Sen viimeisimmän virallisen vakaan version kanssa voi käyttää MySQL:ää, MariaDB:tä, PostgreSQL:ää tai SQLitea. Wordpressin tavoin Drupalin ohjelmointikielenä käytetään PHP:tä. [17]

Drupal on Wordpressiä mutkikkaampi ja raskaampi sisällönhallintajärjestelmä. Se soveltuukin paremmin monimutkaisempien ja laajempien sivujen luomiseen ja hallintaan. Alkuun Drupal voi tuntua kehittäjän näkökulmasta haastavammalta ja kankeammalta kuin Wordpress, mutta perusasioiden selvittyä huomaa, kuinka monimuotoinen sisällönhallintajärjestelmä on kyseessä.

### 3.3 Joomla

Joomla 1.0 julkaistiin vuonna 2005. Kehitystyöhön osallistui suuri joukko Joomla-yhteisön jäseniä ja hyvin sitoutunut ydinkehittäjäryhmä. Sittenmin Joomla on kehittynyt yhdeksi suosituimmaksi sisällönhallintajärjestelmäksi ja se on voittanut myös palkintoja [18].

Uusin versio Joomlaista on 3.3, joka on erittäin monipuolinen ja taipuisa sisällönhallintajärjestelmä. Versio 3.3 tukee MySQL:ää, SQL Serveriä ja PostgreSQL:ää. Palvelimelta Joomla vaatii Apachen, Nginxiin tai Microsoft IIS:än [19].

Vertailtaessa sisällönhallintajärjestelmiä Joomla sijoittuu Drupalin ja Wordpressin välimaastoon. Joomlaissa yhdistyy helppokäyttöisyys ja tehokkuus.

### 3.4 Concrete5

Kolmen edellä mainitun tapaan Concrete5 on niin ikään avoimeen lähdekoodin perustuva sisällönhallintajärjestelmä, joka käyttää MySQL-tietokantaa ja PHP-ohjelmointikieltä. MySQL:n ja PHP:n lisäksi Concrete5 vaatii palvelinohjelmistoksi joko Apachen ja IIS:n. Viralliset dokumentit kuitenkin suosittelevat Apachea. [20] Concrete5 tarjoaa hyvin monipuolisen ja varsinkin loppukäyttäjäystävällisen vaihtoehdon. Concrete5 julkaistiin vuonna 2008 ja on vuoden 2014 marraskuuhun mennessä kehittynyt versioon 5.7. [21]

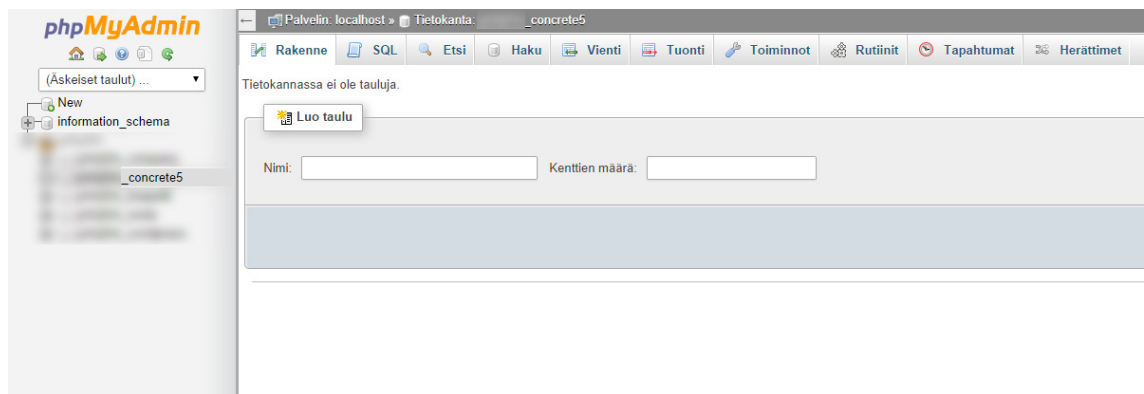
Loppukäyttäjän näkökulmasta Concrete5:n ehdoton etu on sen tarjoama käyttöliittymä sivujen päivittämiseen. Näistä neljästä sisällönhallintajärjestelmästä Concrete5 on ehdottomasti nuorin ja se taas asettaa haasteita kehittäjälle. Wordpressin, Joomla:n ja Drupalin tavoin Concrete5:llä ei ole yhtä laajaa tuki- ja kehittäjäverkostoa. Kehittäjän perspektiivistä tämä lisää haasteita ja kehittäminen voi näin olla hitaampaa, mutta samalla palkitsevampaa.

### 3.5 Julkaisujärjestelmän valinta

Tähän opinnäytetyöhön valittiin Concrete5, koska se tarjoaa juuri loppukäyttäjälle mahdollisimman helpon ja luontevan tavan päivittää sivustoa. Myös se että julkaisujärjestelmänä Concrete5 on hieman tuntemattomampi, puoltaa sen valintaa tarkempaan tarkasteluun.

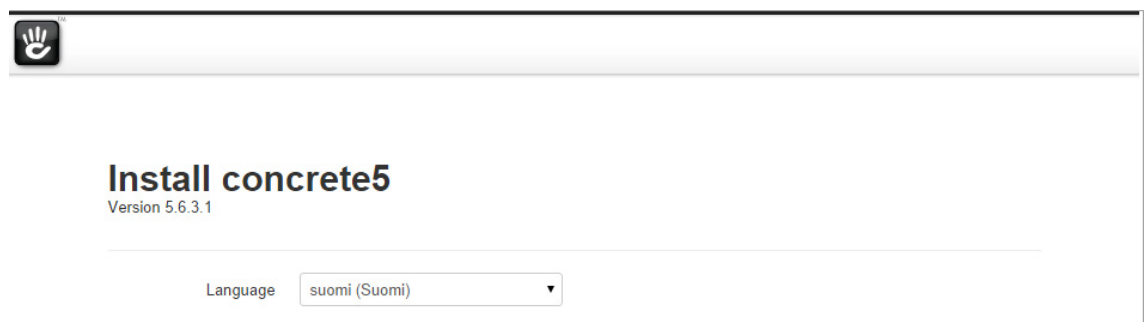
## 4 CONCRET5:N ASENNUS

Concrete5:stä julkaistiin 12.9.2014 versio 5.7. Tähän työhön valittiin kuitenkin versio 5.6.3.1. Opinnäytetyötä varten käytössä on palvelin, johon Concrete5 voidaan asentaa. Phpmyadmin-hallintapaneelin kautta Concrete5:tä varten on luotu tyhjä tietokanta. Turvallisuussyistä tarkat osoitteet on sumennettu kuvista.



Kuva 5. Tyhjä tietokanta sisällönhallintajärjestelmää varten

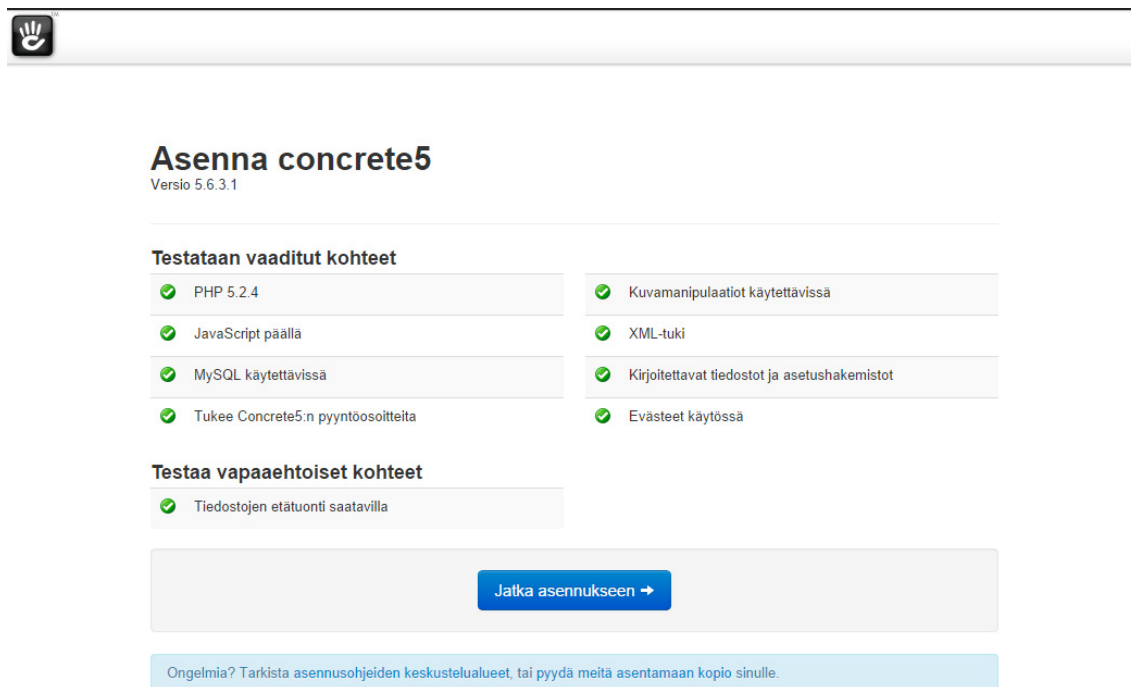
Tietokantaan luotiin myös käyttäjä täysillä oikeuksilla. Palvelimelle lisättiin uusi hakemisto, johon Concrete5:en asennuspaketti siirrettiin FileZilla FTP-ohjelmalla. Kun asennuspaketti on halutussa sijainnissa ja selaimen osoiteriville syöttää palvelimella olevan hakemiston sijainnin, avautuu Concrete5:en asennusruutu. Ensimmäisessä osassa valitaan haluttu kieli.



Kuva 6. Kielivalinta


Kielivalinnan jälkeen asennus käy läpi palvelimelta vaaditut kohteet ja varmistaa, että Concrete5 on mahdollista asentaa ko. palvelimelle.

Asennusohjelma käy läpi seuraavat kohdat: palvelimelta löytyy riittävä PHP-tuki, JavaScript on päällä, MySQL on käytettävissä, palvelin tukee Concrete5:n pyyntöosoitteita, kuvamanipulaatiot ovat päällä, saatavana on XML-tuki, hakemistojen ja tiedostojen oikeudet on asetettu oikein ja evästeet ovat käytössä. Mikäli jokin näistä kohdista ei toimi, asennusta ei voida jatkaa.



Kuva 7. Vaatimusten testaus

Viimeisenä Concrete5:n asennusohjelma pyytää tiedot sivusta ylläpitäjästä ja tietokannasta. Asennuksen voi ajaa esimerkkisisällön kanssa, mutta tässä työssä asensimme tyhjän testisivun. ”Asenna concrete5”-nappia painamalla sisällönhallintajärjestelmä asentuu palvelimelle.



## Asenna concrete5

Versio 5.6.3.1

### Sivuston tiedot

Sivuston nimi:

### Tietokannan tiedot

Palvelin:

MySQL:n käyttäjätunnus:

MySQL:n salasana:

Tietokannan nimi:


### Ylläpitäjän tiedot

Sähköpostiosoite:


Salasana:

Vahvista salasana:

### Esimerkkisisällöt

☐


**Esimerkkisisältö blogilla.**  
Hyvä lähtökohta houkuttelevalle www-sivustolle blogilla.

☒


**Tyhjä sivusto**  
Asenna ainoastaan kohteet, joita concrete5 vaatii. Tämä luo tyhjän sivuston.

concrete5 veteraanit voivat valita "Tyhjä sivusto", mutta muutoin suosittelemme aloittamaan esimerkkisisällön avulla.

[Asenna concrete5](#)

Kuva 8. Sivun, tietokannan ja ylläpitäjän tiedot

## 5 RESPONSIIVISEN POHJAN HYÖDYNTÄMINEN SISÄLLÖNHALLINTAJÄRJESTELMÄSSÄ

### 5.1 Toimeksiannon tavoite

Opinnäytetyöhön kuului käytännön työnä responsiivisen pohjan hyödyntäminen sisällönhallintajärjestelmässä. Toimeksianto tehtiin Mobiilimarkkinointi Routa Oy:n kanssa, jossa toimin web-designerina.

Pohjaa ei tehty varta vasten tiettyä asiakasprojektia varten, vaan se toimii pikemminkin pohjana jatkokehitystä varten ja uusien ideoiden testipohjana. Toimeksiannon aloituspalaverissa ei sovittu tarkkoja linjauksia pohjan rakenteesta vaan suunnitteluun annettiin vapaat kädet.

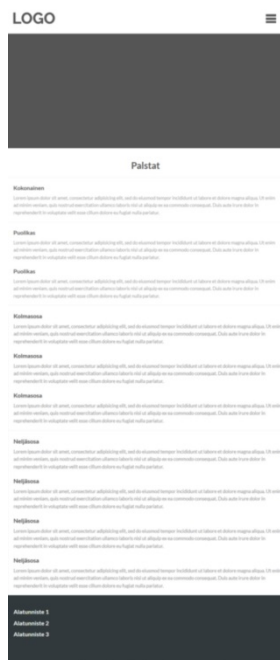
Lähtökohtana oli rakentaa selkeä ja rakenteeltaan yksinkertainen pohja, jossa ei ole monimutkaista tyylittelyä, jotta jatkokehitys olisi mahdollisimman helppo aloittaa. Tavoitteena oli syventää omaa tietämystä responsiivisesta verkkosuunnittelusta sekä tutustua ja omaksua uusi julkaisujärjestelmä, jonka hallitsemisesta on myöhemmin hyötyä työtehtävissä.

### 5.2 Responsiivinen pohja

Jokainen verkkosivuprojekti on ainakin jossain määrin erilainen, joten responsiivisesta pohjasta ei ole tehty yksityiskohtaista tai valmista sivua, vaan siihen on luotu perusrakenteet ja yleisesti kaikilta verkkosivuilta löytyvä alueet sekä elementit. Se toimii pohjana mahdolliselle jatkokehitykselle. Pohjaan valittiin ainoastaan yksi media queryillä toteutettu breakpoint, joka määriteltiin kohtaan alle 768 pikseliä. Tässä kohdassa palstojen leveys vaihtuu alle 768 pikseliä leveillä näytöillä 100-prosenttisiksi ja navigaatio muutetaan mobiilimuotoon.



Kuva 9. Responsiivinen pohja yli 768 pikseliä leveillä näytöillä



Kuva 10. Responsiivinen pohja alle 768 pikseliä leveillä näytöillä.

Concrete5-sivusto koostuu joukosta php-tiedostoja. Opinnäytetyötä varten luodussa pohjassa on ainoastaan seuraavat tiedostot header.php, footer.php ja default.php. Uusia sivutyyppejä varten voidaan luoda default.php:n lisäksi muita sivupohjia. JavaScriptin ja CSS-tyylitiedostojen lisääminen onnistuu viittaamalla niihin sivuston header- tai footer-tiedostossa.

Esimerkkikoodi 1:

```
<script                                type="text/javascript"                                src="<?=$view-
>getThemePath()?>/js/javascript.js"></script>
```

Esimerkkikoodi 1:ssä kerrotaan ensin järjestelmälle, että lisättävä elementti sisältää JavaScript-koodia. Tämän jälkeen haetaan pohjateeman tiedostopolku ja kerrotaan, että javascript.js sijaitsee kansiossa js.

Tyylitiedoston linkittäminen teempohjaan onnistuu samankaltaisesti. Esimerkkikoodissa 2 haetaan tyyli.css theemapohjan kansioista styles.

Esimerkkikoodi 2

```
<link    rel="stylesheet"    href="<?php    echo    $this->getThemePath();
?>/styles/tyyli.css" />
```

Concrete5:llä on muutamia sille ominaisia määrittämiä, jotka pitää tehdä ennen teemapohjan toimimista. Teemapohjan header- ja footer-tiedostot sijoitetaan elements-kansioon. Sivutyyppeiden php-tiedostot sijaitsevat sen sijaan teemakansion juuressa. Kaikkien php-tiedostojen ylimmäksi riviksi asetetaan seuraava koodi: `<?php defined('C5_EXECUTE') or die("Access Denied."); ?>`. Tällä varmistetaan, että kaikki php-kutsut tulevat Concrete5-järjestelmän sisältä. Samalla varmistetaan, että koodi on validia eikä se avaa turvallisuusuhkia.

Header.php-tiedostoon on lisättävä vielä seuraava koodi: `<?php Loader::element('header_required') ?>`. Tällä mahdollistetaan Concrete5:n oikeus kyseiseen tiedostoon. Footer.php -tiedostoon asetetaan myös vastaava koodi, jossa header-sana on korvattu footer-sanalla. [22]

Lukuun ottamatta näitä kolmea poikkeusta Concrete5:n teemapohjan rakenne noudattelee tavallisen html-dokumentin runkoa.



### 5.2.1 Ylätunniste

Pohjan ylätunniste sisältää paikat logolle sekä päävalikolle. Ylätunnisteeseen on helppo projektin vaatiessa lisätä muitakin alueita, mutta logo ja päävalikko löytyvät lähes poikkeuksetta joka sivustolta. Logon paikka on ylätunnisteen vasemmassa reunassa, ja päävalikko on sijoitettu oikeaan reunaan.

### 5.2.2 Sisältöalue

Sisältöalueelle on luoto viisi erikokoista palsta-aluetta, ja niihin on havainnollistamiseksi asetettu esimerkkisisältöä. Palstojen leveydet ovat 100 %, 50%, 33% ja 20%. Alle 768 pikseliä leveillä näytöillä kaikkien palstojen leveys asetetaan 100-prosenttiseksi. Koko sisältöalue on wrapper-luokan sisällä ja tämän luokan maksimi leveydeksi on asetettu 1 260 pikseliä. Sisältöalueen kuville on asetettu leveydeksi 100% ja korkeudeksi automaattinen.

Kuvat säilyttävät oikean kuvasuhteensa ja skaalautuvat sen elementin mukaan, johon ne on asetettu. Mikäli responsiivista pohjaa hyödynnettäisiin projektissa, jossa kuvia olisi paljon, sivulla käytettäisiin HiSRC:tä tai jotain vastaavaa ratkaisua.

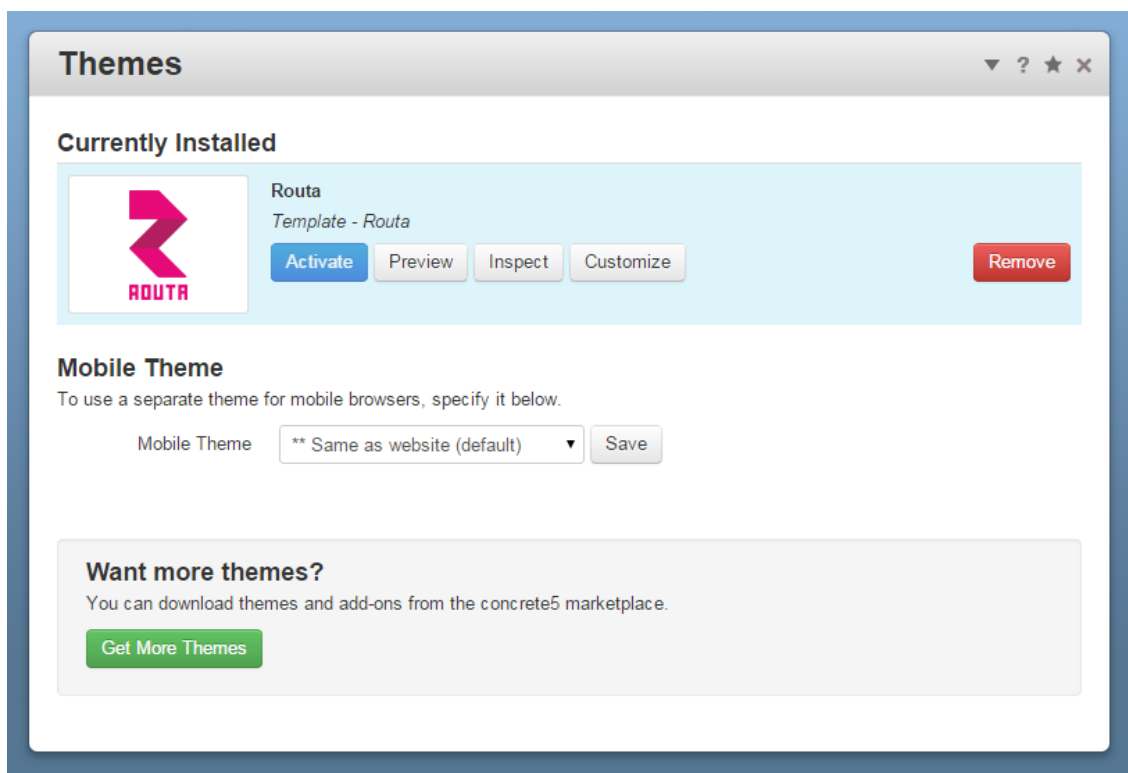
Pohjan tekstikoot on asetettu seuraavasti: H1-elementti on 1,6 rem-yksikköä, H2-elementti 1 rem-yksikköä ja tavallinen teksti eli p-elementti on 0,8 rem-yksikköä. Yli 768 pikselin leveydellä html-dokumentin teksti koko on 18 pikseliä ja kun ruutukoko on alle 768 pikseliä, niin html-dokumentin tekstikokoa pienennetään 16 pikseliin.

### 5.2.3 Alatunniste

Alatunnisteessa on kolme palstaa, joiden leveys on 33%. Päätelaitteen ruutukoon leveyden ollessa alle 768 pikseliä palstojen leveydeksi asetetaan niin ikään 100%.

### 5.3 Responsiivisen pohjan käyttöönotto

Concrete5:n teemapohjan tiedostoja varten pitää luoda uusi kansio Concrete5:n juurihakmiston themes-kansioon. Tähän kansioon siirretään kaikki teemapohjan tiedostot. Teemapohjan käyttöönotto tapahtuu Concrete5:n hallintapaneelista teemat-valikosta. Teemapohja aktivoidaan painamalla Activate halutun teemapohjan kohdalla.



Kuva 11. Teemapohjan aktivointi

Teemapohjan tyylit ja asetukset tulevat voimaan kaikille sivuston sivuille.

## 6 YHTEENVETO

Responsiivisen suunnittelun tarkasteleminen opinnäytetyönä osoittautui varsin mielekkääksi työksi. Ideana responsiivinen suunnittelu on jo melko vanha. Uusia tekniikoita toteuttaa responsiivisuutta kehitellään kuitenkin jatkuvasti, joten alaa on seurattava tauotta. Opinnäytetyön päämääränä oli tutustua responsiiviseen suunnitteluun ja siihen, miten sitä voidaan hyödyntää sisällönhallintajärjestelmässä. Käytännön osuuteen sisällönhallintajärjestelmäksi valittiin Concrete5. Työtä varten suunniteltiin yksinkertainen responsiivinen pohja, jota käytiin samalla esimerkkinä.

Opinnäytetyö onnistui suhteellisen hyvin, ja tavoite tutustua responsiiviseen suunnitteluun ja sen hyödyntämiseen sisällönhallintajärjestelmässä saavutettiin. Työ ei ainakaan tällaisenaan päädy tuotantokäyttöön, mutta se antoi perspektiiviä tuleviin projekteihin ja avasi omaa käsitystä verkkosuunnittelusta ja sen tilasta tällä hetkellä.

Vaikka idea responsiivisesta suunnittelusta keksittiin useita vuosia sitten, edelleen verkosta tuntuu löytyvän suuri määrä yrityksiä, jotka eivät huomioi tätä seikkaa sivustoillaan ollenkaan. Markkinoille tulee jatkuvasti uusia laitteita, ja käyttäjien tottumukset vaihtuvat ja muokkautuvat eri laitteiden myötä. Responsiivinen suunnittelu ei ole täysin yhteensopiva tulevaisuuden kanssa, mutta tällä hetkellä se on tehokkain tapa palvella verkkosivun käyttäjiä mahdollisimman monipuolisesti päätelaitteesta riippumatta.

## LÄHTEET

- [1] The Man in Blue 2004. Viitattu 22.8.2014  
<http://www.themaninblue.com/writing/perspective/2004/09/21/>
- [2] A List Apart 2010. Responsive Web Design. Viitattu 5.9.2014  
<http://alistapart.com/article/responsive-web-design>
- [3] Gartner Oy 2013. Gartner Says Worldwide PC, Tablet and Mobile Phone Combined Shipments to Reach 2.4 Billion Units in 2013. Viitattu 23.8.2014  
<http://www.gartner.com/newsroom/id/2408515>
- [4] LaGrone, B. 2013. HTML5 and CSS3 Responsive Web Design Cookbook, Birmingham: Packt Publishing, 115 s.
- [5] Google 2014. Building Smartphone-Optimized Websites. Viitattu 23.8.2014  
<https://developers.google.com/webmasters/smartphone-sites>
- [6] Gasston, P. 2011. The Book of CSS3: A Developer's Guide to the Future of Web Design, San Francisco: No Starch Press, Inc, 252 s.
- [7] Robbins, J. 2012. Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics. Sebastopol: O'Reilly Media, 56 s.
- [8] Grannell, C.; Sumner, V. & Synodinos, D. 2012. The Essential Guide To HTML5 and CSS3 Web Design. New York: Apress and friends of ED, 26 s.
- [9] Wikipedia 2014. Em (typography). Viitattu 3.9.2014  
[http://en.wikipedia.org/wiki/Em\\_%28typography%29](http://en.wikipedia.org/wiki/Em_%28typography%29)
- [10] Nielsen Norman Group 2010. Photos as Web Content. Viitattu 30.8.2014  
<http://www.nngroup.com/articles/photos-as-web-content/>
- [11] GitHub 2014. HiSRC. Viitattu 2.10.2014 <https://github.com/teleject/hisrc>
- [12] Frain, B. 2012. Responsive Web Design with HTML5 and CSS3, Birmingham: Packt Publishing, 36 s.
- [13] Wikipedia 2014. Content management system. Viitattu 19.10.2014  
[http://en.wikipedia.org/wiki/Content\\_management\\_system](http://en.wikipedia.org/wiki/Content_management_system)
- [14] W3Techs 2014. W3Techs - World Wide Web Technology Surveys. Viitattu 3.10.2014  
<http://w3techs.com/>
- [15] Wordpress 2014. About Wordpress. Viitattu 3.10.2014 <http://wordpress.org/about/>
- [16] Drupal 2014. About Drupal. Viitattu 4.10.2014 <https://www.drupal.org/about>
- [17] Drupal 2014. System requirements. Viitattu 4.10.2014 <https://www.drupal.org/requirements>
- [18] Marriot, J. & Waring E. The Official Joomla! Book. Boston: Pearson Education, 2011, 3 s.
- [19] Joomla 2014. Technical Requirements. Viitattu 5.10.2014 <http://www.joomla.org/technical-requirements.html>
- [20] Concrete5 2014. System Requirements. Viitattu 5.10.2014  
[https://www.concrete5.org/documentation/background/system\\_requirements/](https://www.concrete5.org/documentation/background/system_requirements/)

[21] Concrete5 2014. History. Viitattu 5.10.2014  
[https://www.concrete5.org/about/our\\_philosophy/history](https://www.concrete5.org/about/our_philosophy/history)

[22] Laubacher, R. 2013. Creating Concrete5 Themes, Birmingham: Packt Publishing, 53 s.